

Fig. 1

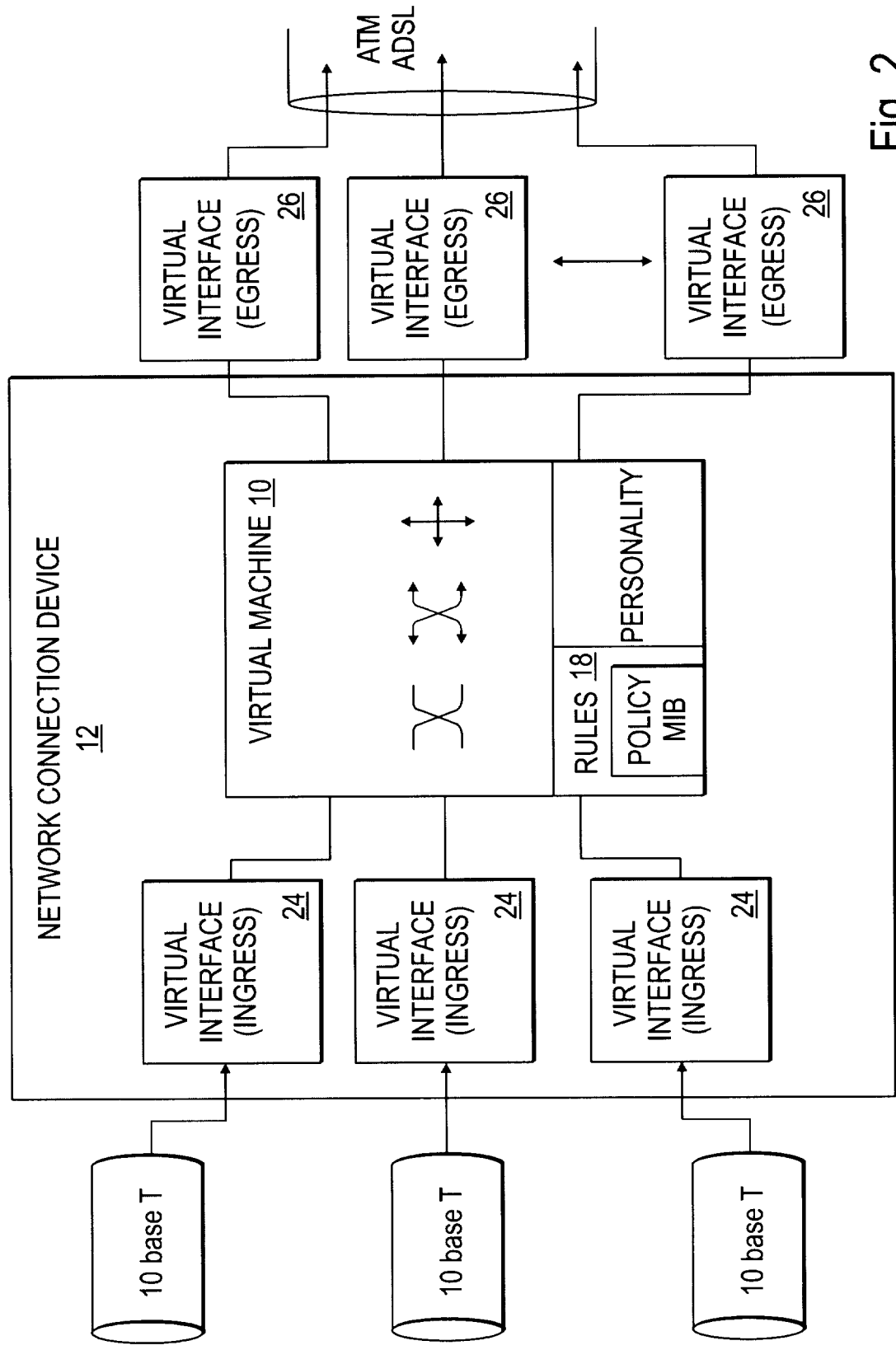


Fig. 2

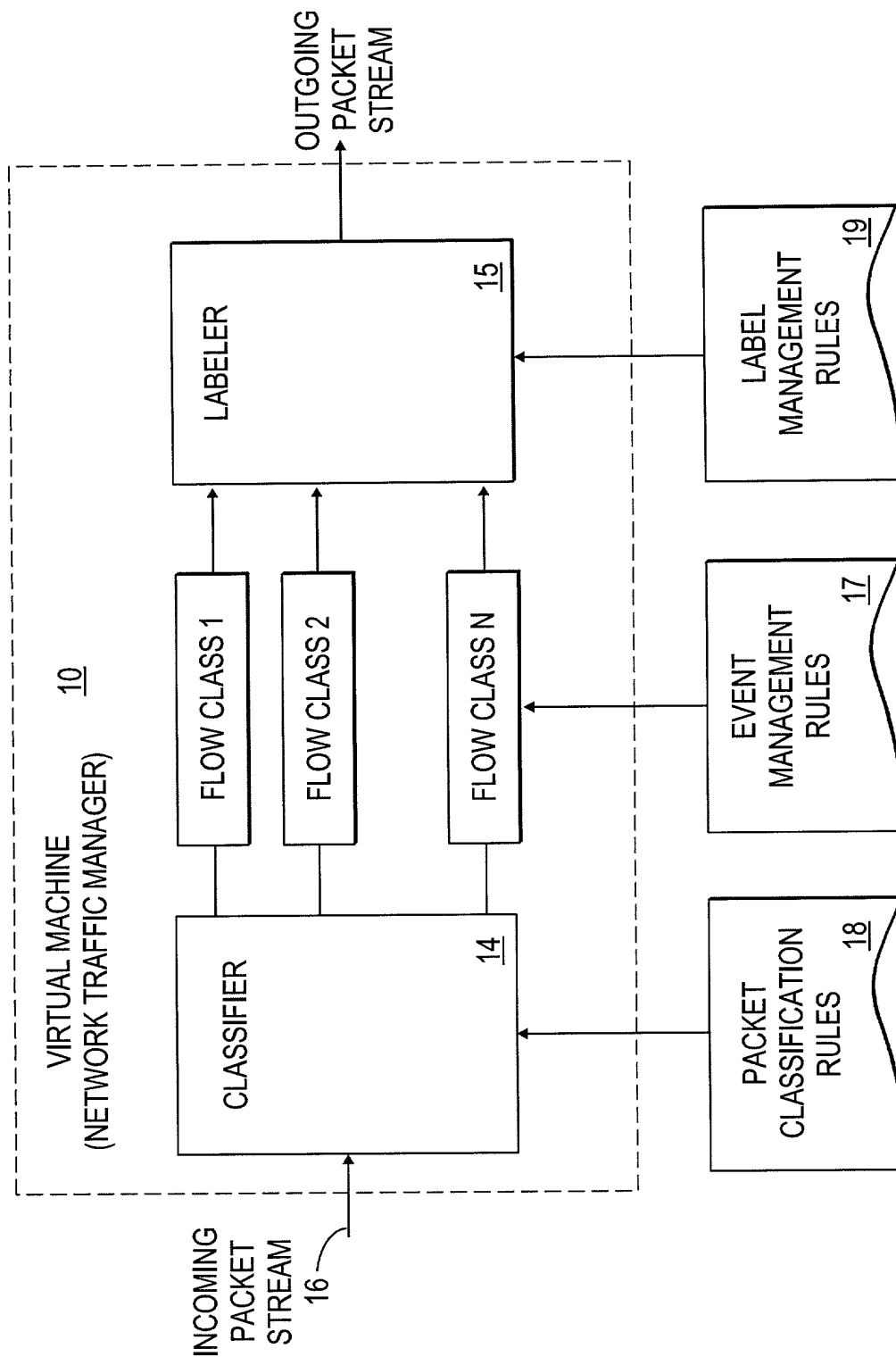


Fig. 3

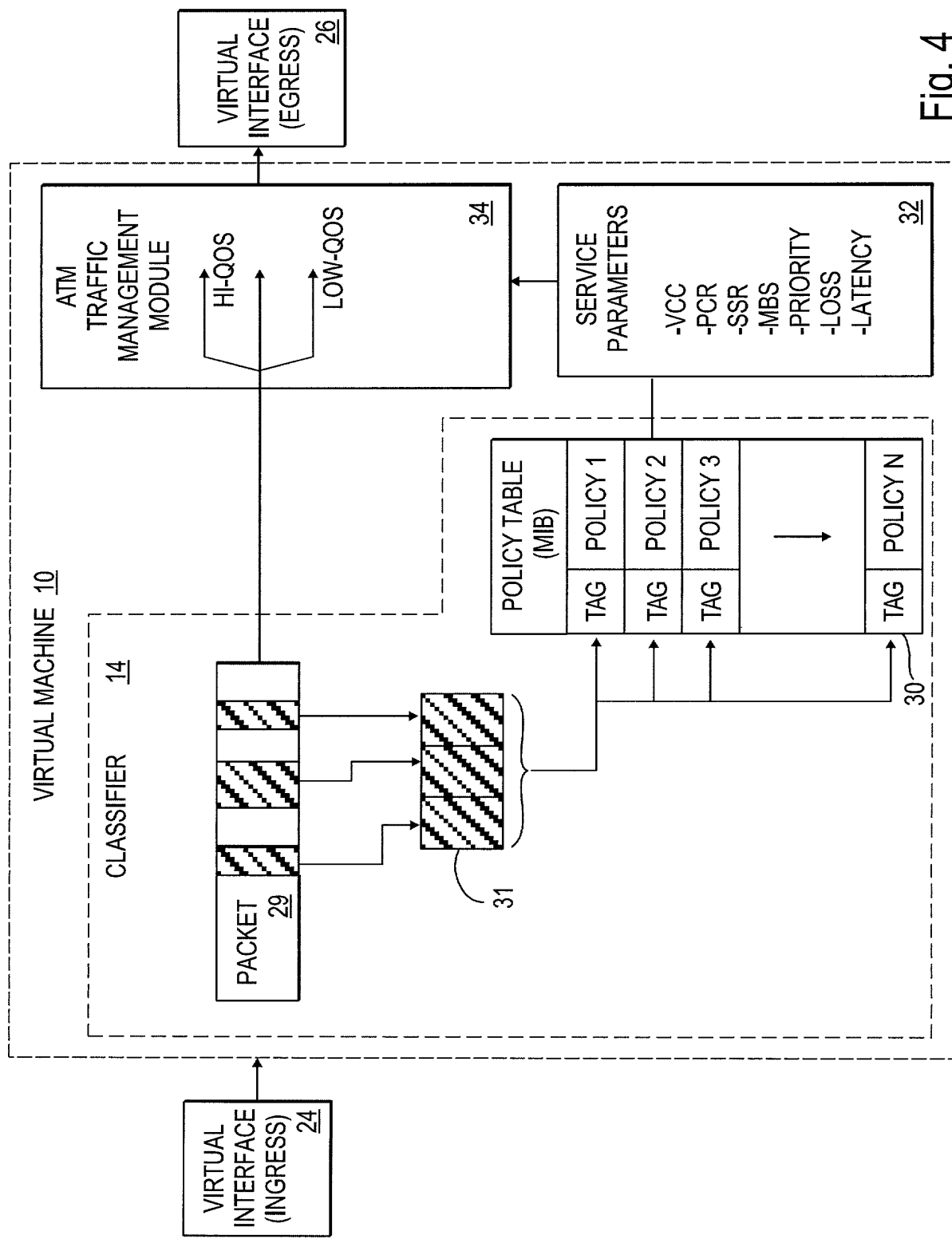


Fig. 4

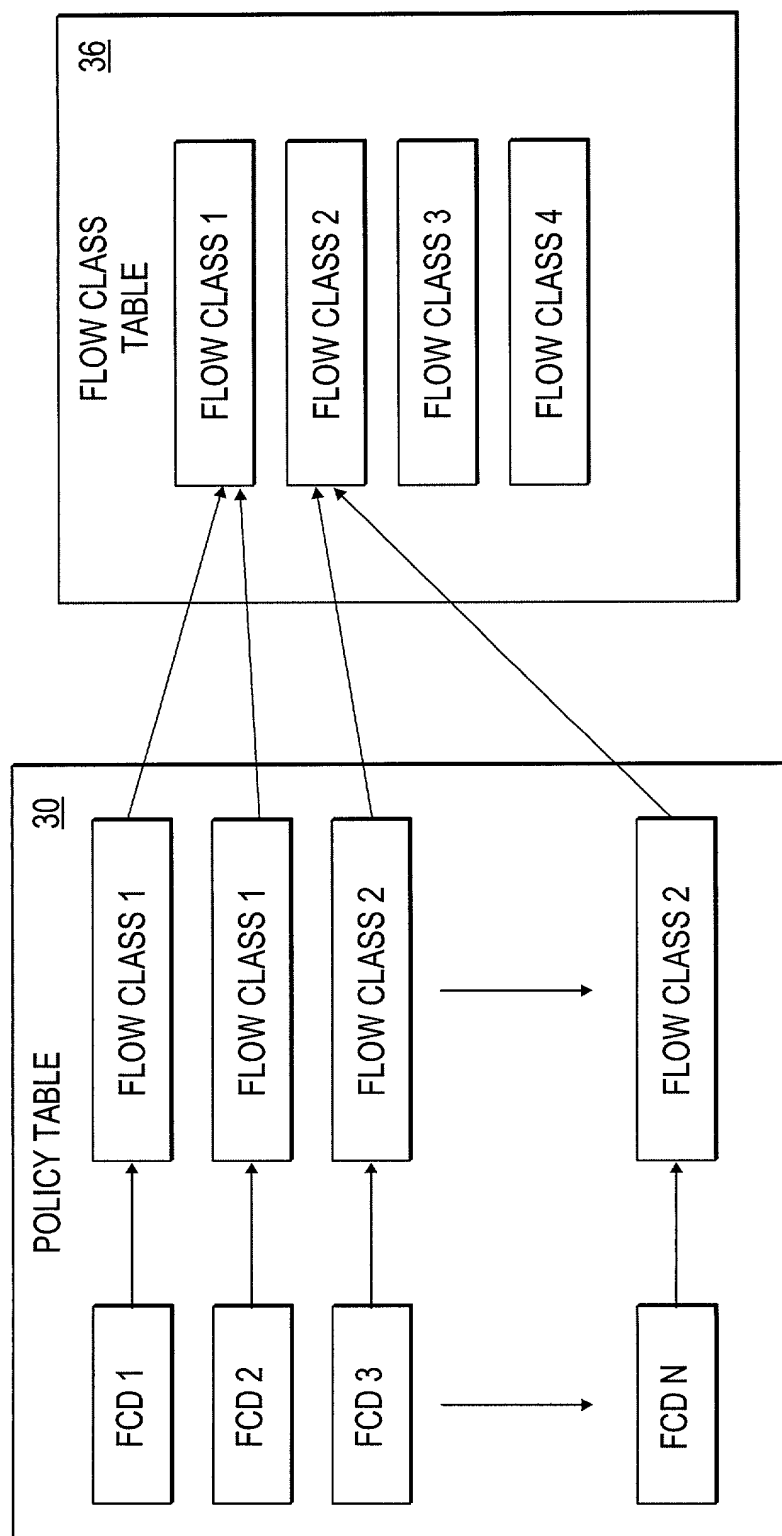


Fig. 5

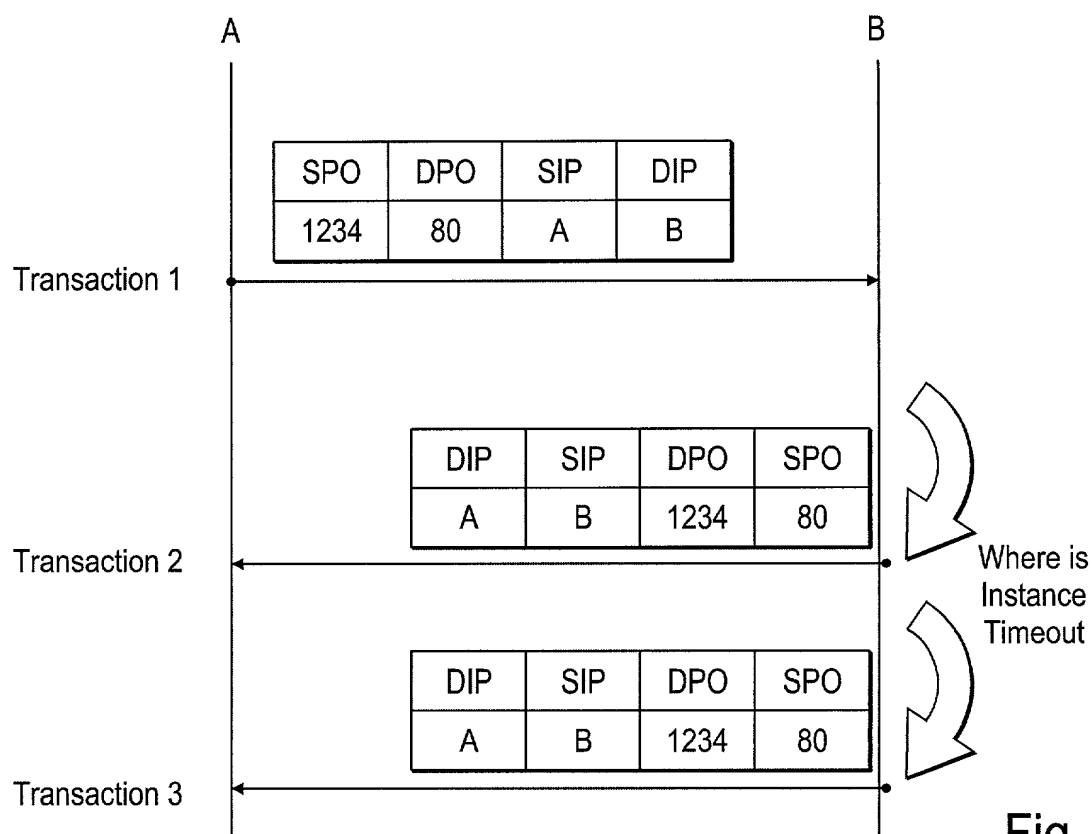


Fig. 6

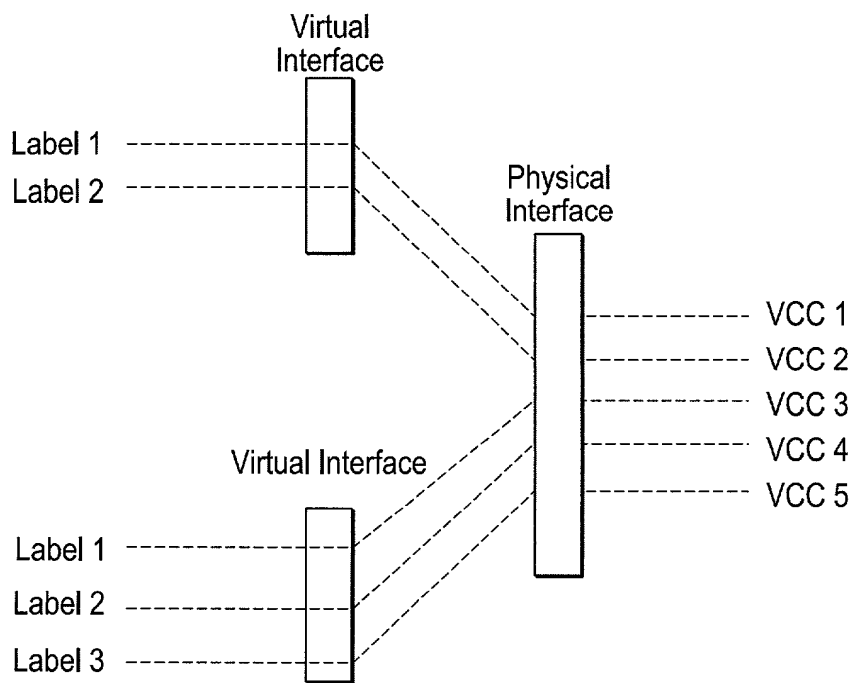


Fig. 7

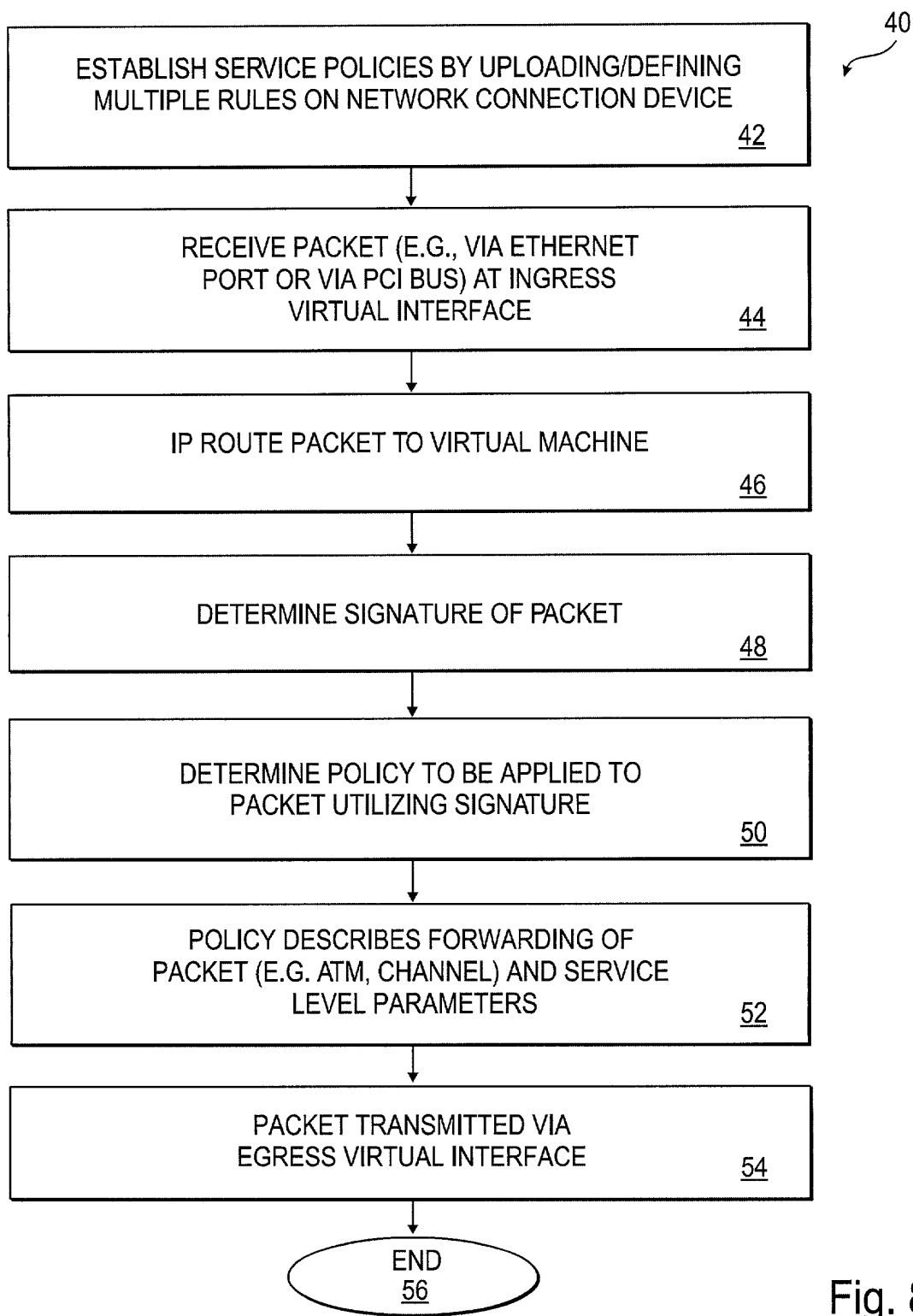


Fig. 8

Fig. 9 is a block diagram of a system for generating a rule program (binary object) from an operations file and a rule file. The system includes an operations file 62, a rule file 64, a virtual machine compiler 60, and a rule program (binary object) 66. The operations file 62 and the rule file 64 are inputs to the virtual machine compiler 60. The virtual machine compiler 60 outputs the rule program (binary object) 66, which is then used by a virtual machine.

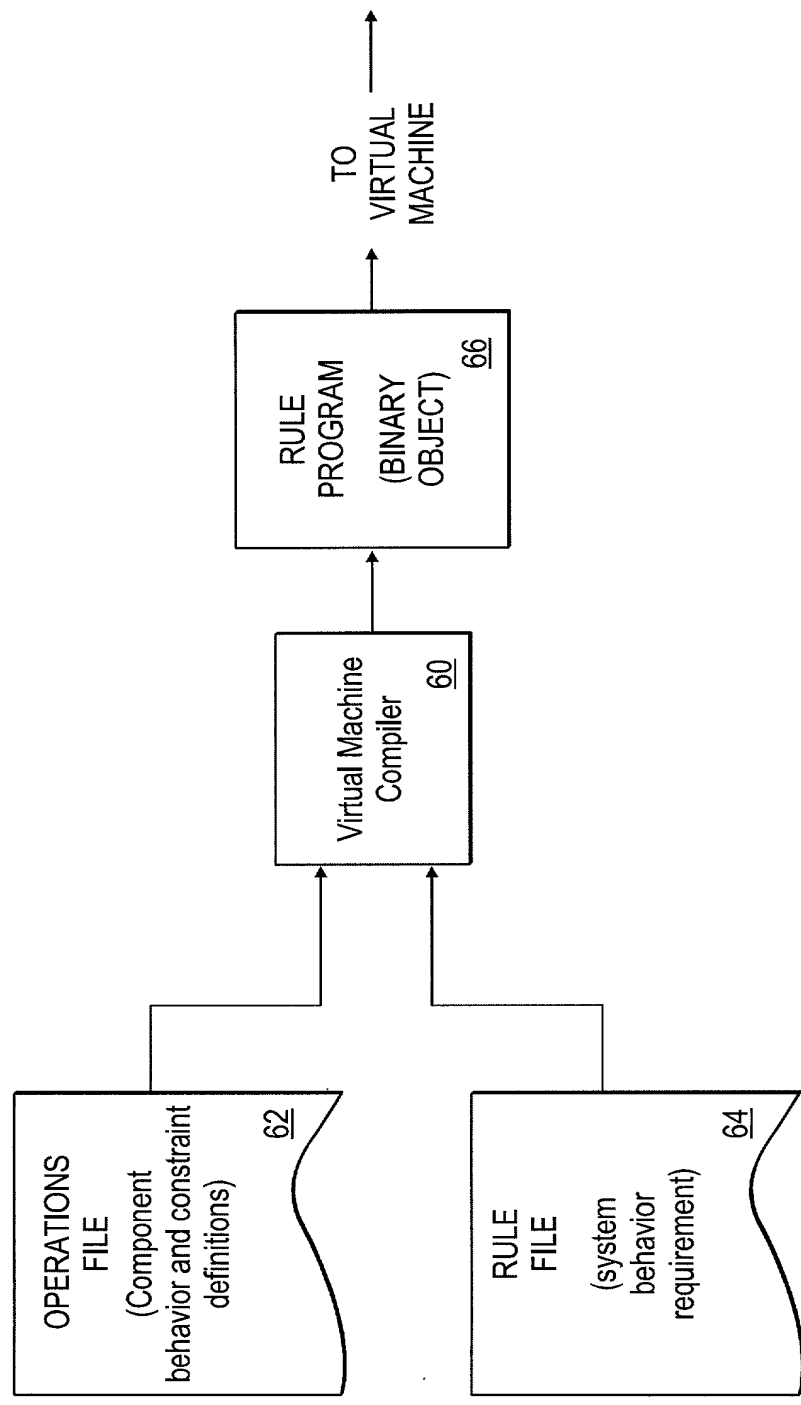


Fig. 9



FIG. 10 is a block diagram of a system architecture showing the interaction between a Process, a Rule Program, and a Context. The Process (70) contains Component A (70) and Component N (73). The Rule Program (66) contains Rule (Sequence) (68) and Rule (68). The Context (75) contains Component A (72) and Component N. Arrows indicate the flow of data and control between these components.

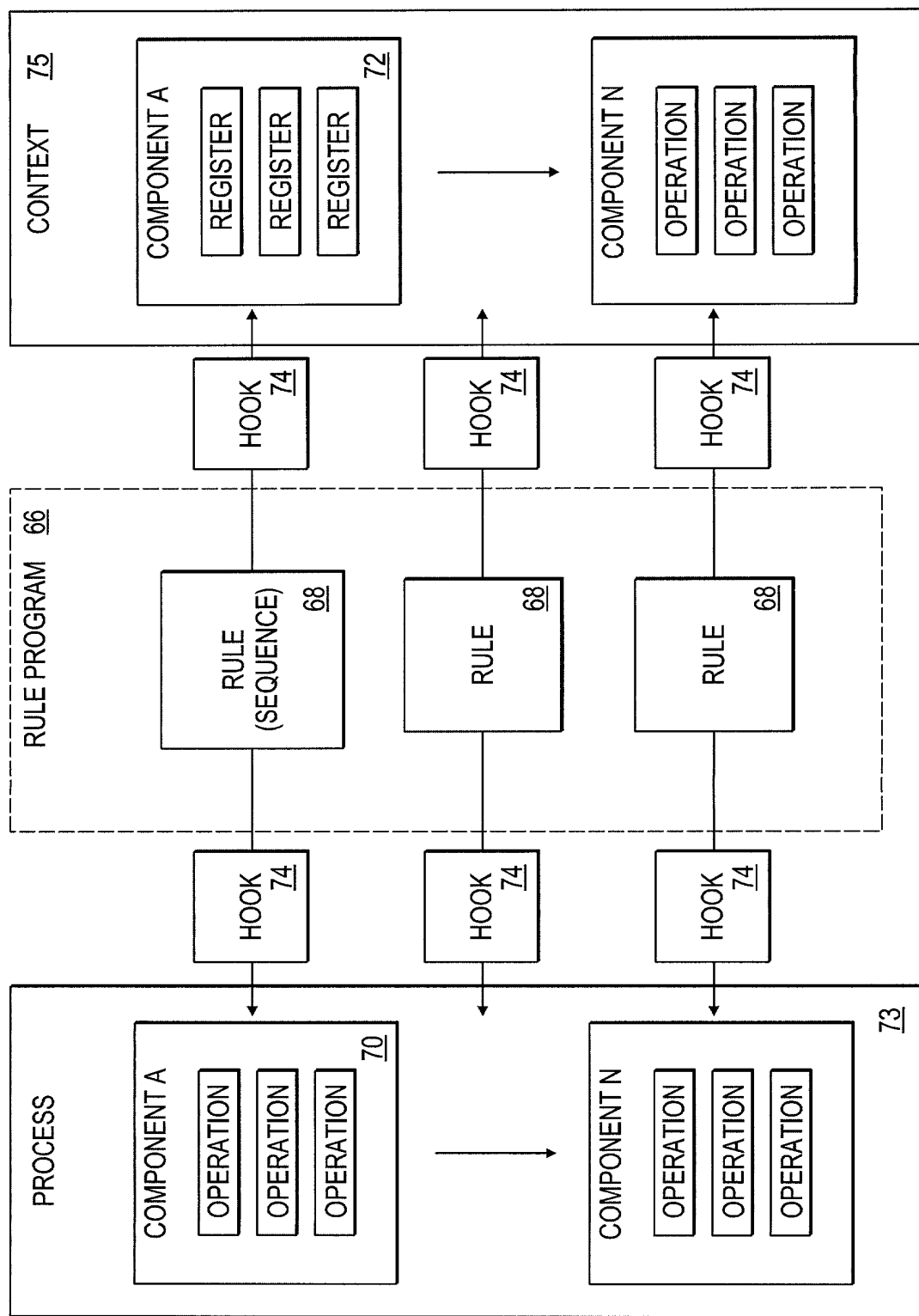


Fig. 10

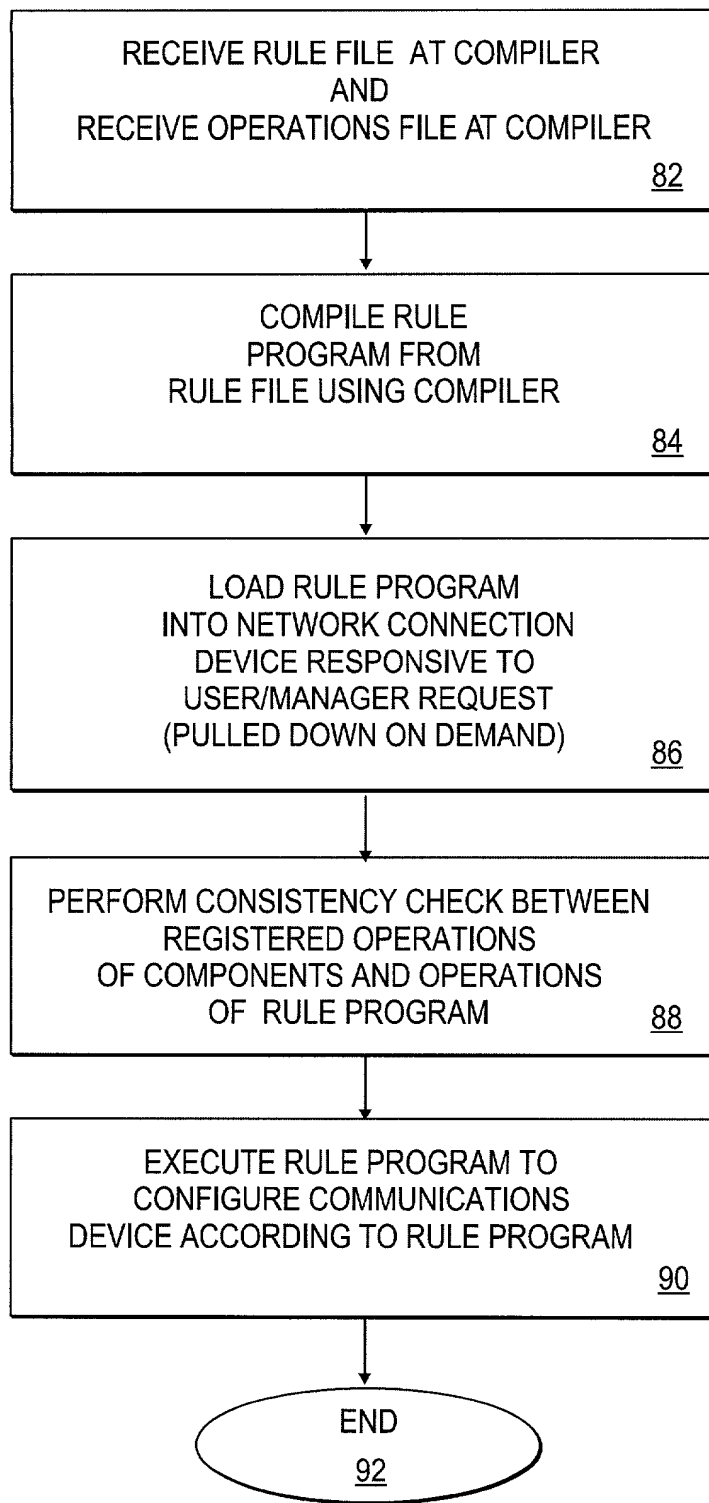


Fig. 11

FIG. 12 is a block diagram of a network architecture. The architecture includes two workstations, Workstation A 102 and Workstation B 102, connected to a central LAN 104. Each workstation contains a (VNIC) CLIENT APPLICATION 100. User A 106 is connected to Workstation A 102, and User B 106 is connected to Workstation B 102. The LAN 104 is connected to a NETWORK CONNECTION DEVICE (E.G., SWITCH, BRIDGE, ROUTER) 12. This device contains a VIRTUAL MACHINE 10, which is connected to A INFORMATION PROFILE 108 and B INFORMATION PROFILE 108. The NETWORK CONNECTION DEVICE 12 is also connected to a network represented by a cylinder with three arrows pointing outwards.

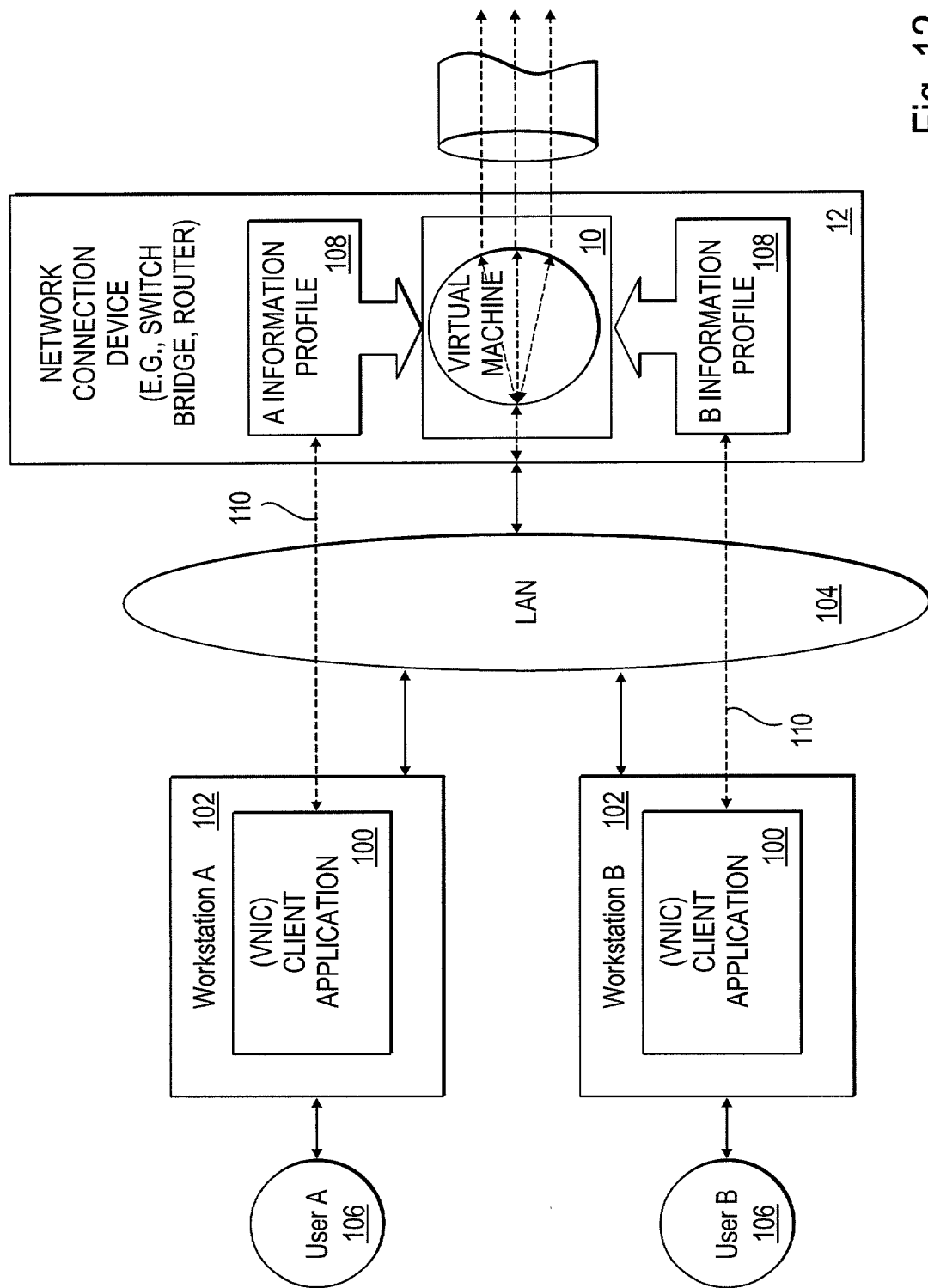


Fig. 12

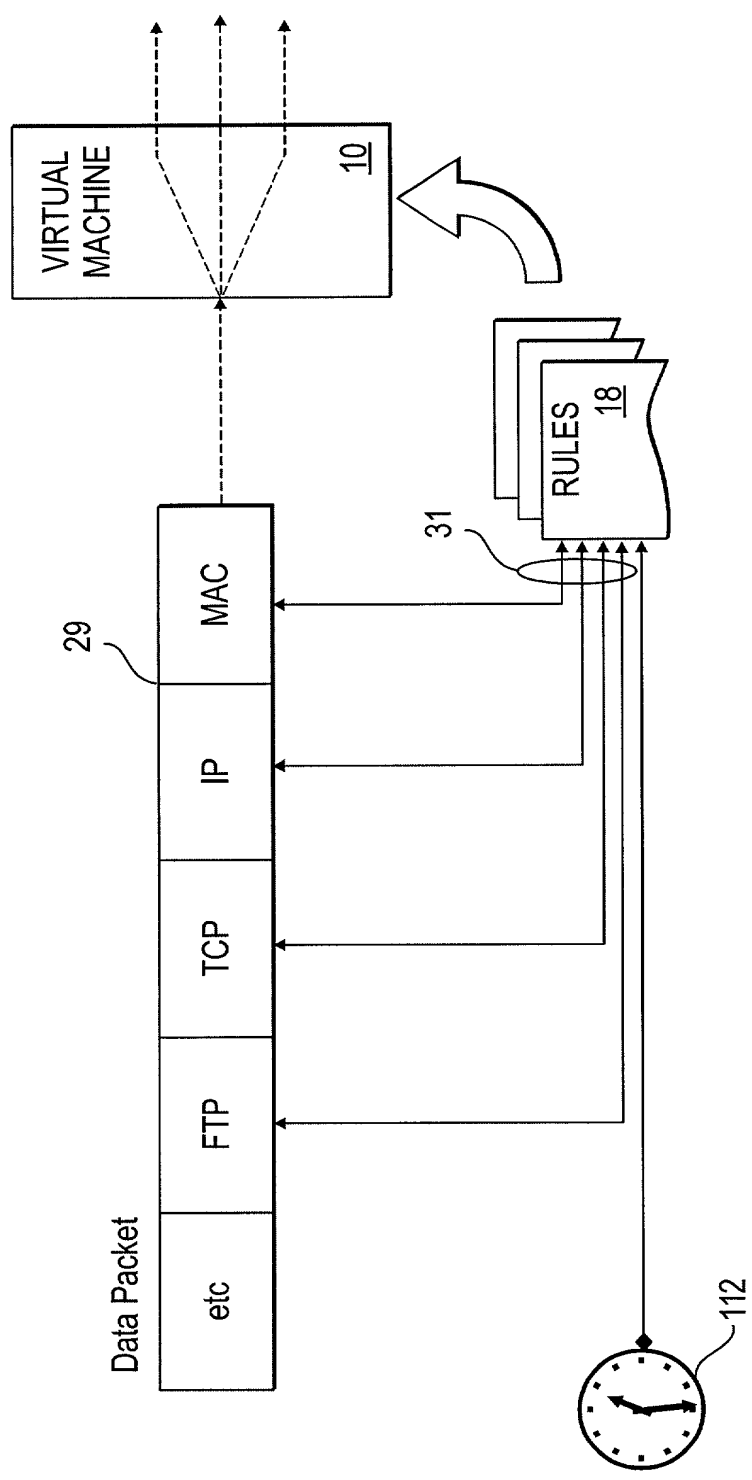


Fig. 13

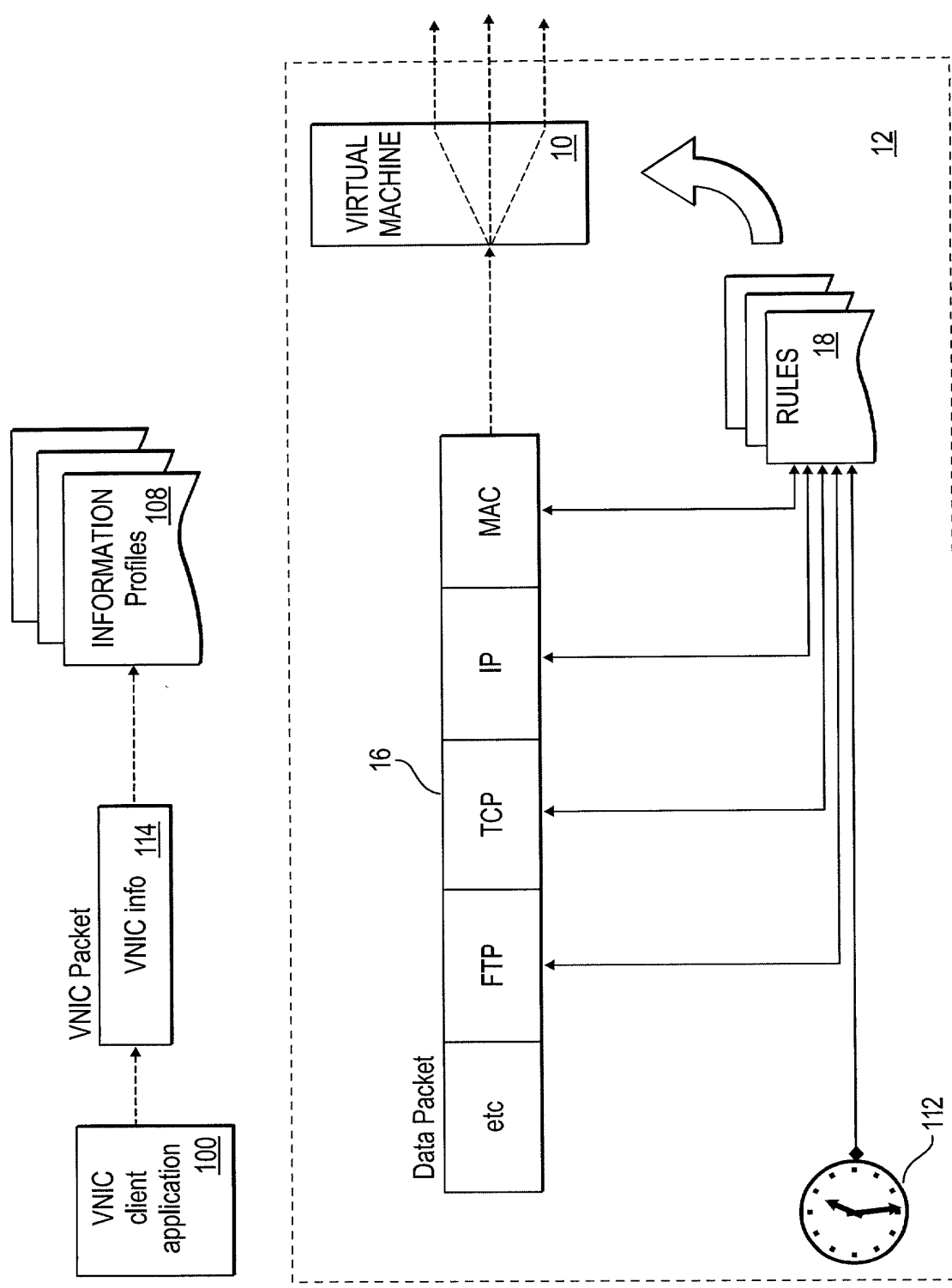


Fig. 14

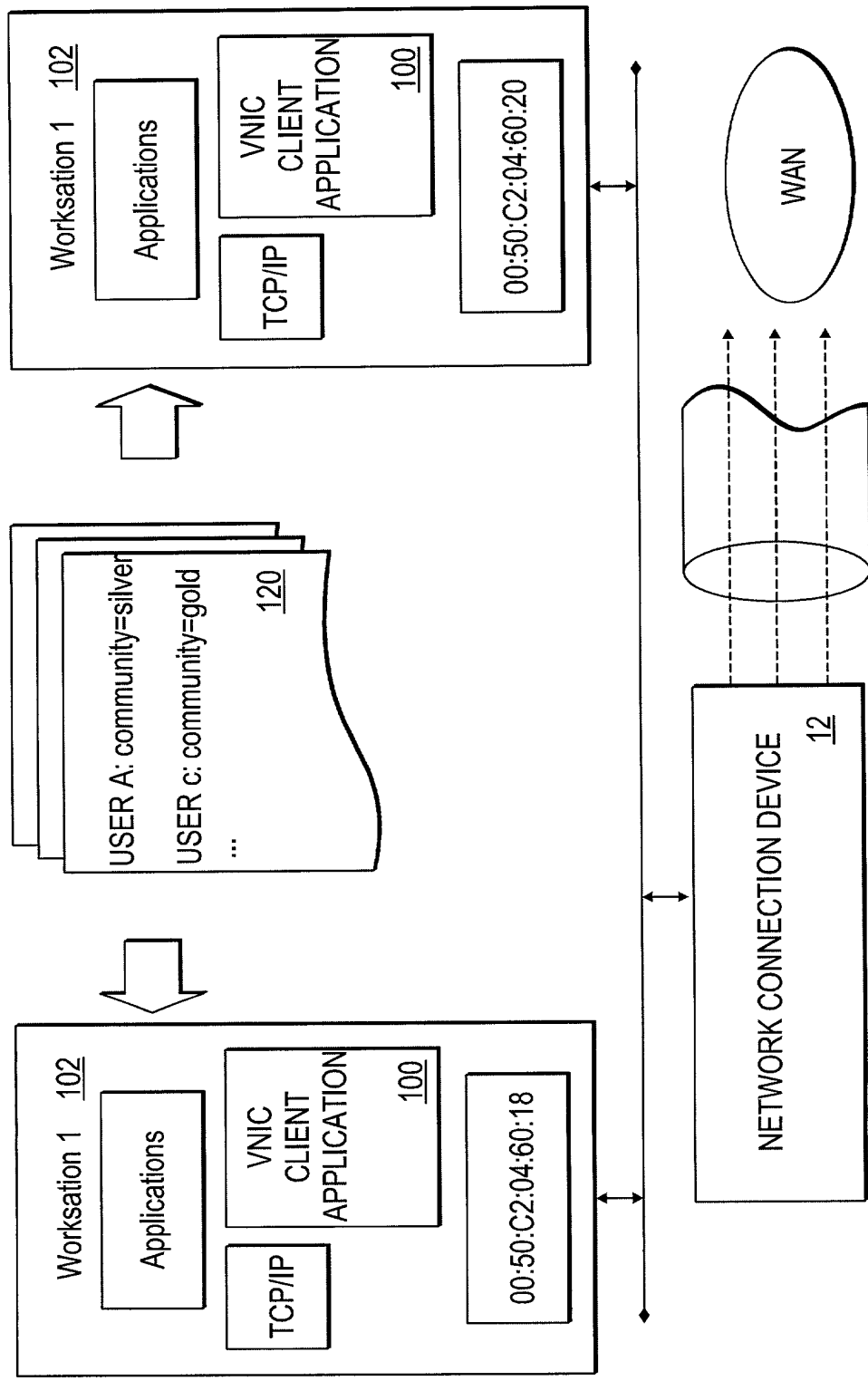


Fig. 15

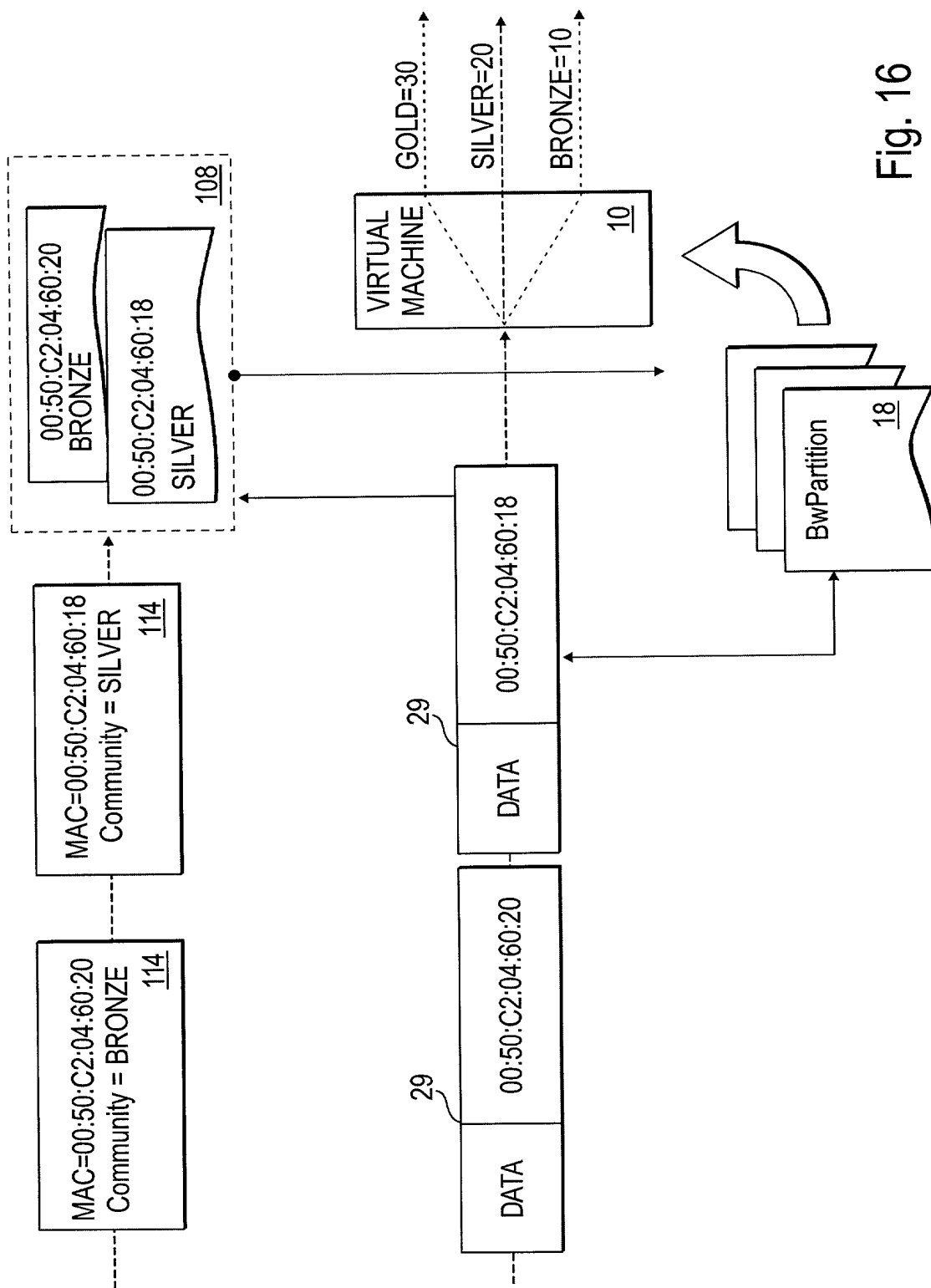


Fig. 16

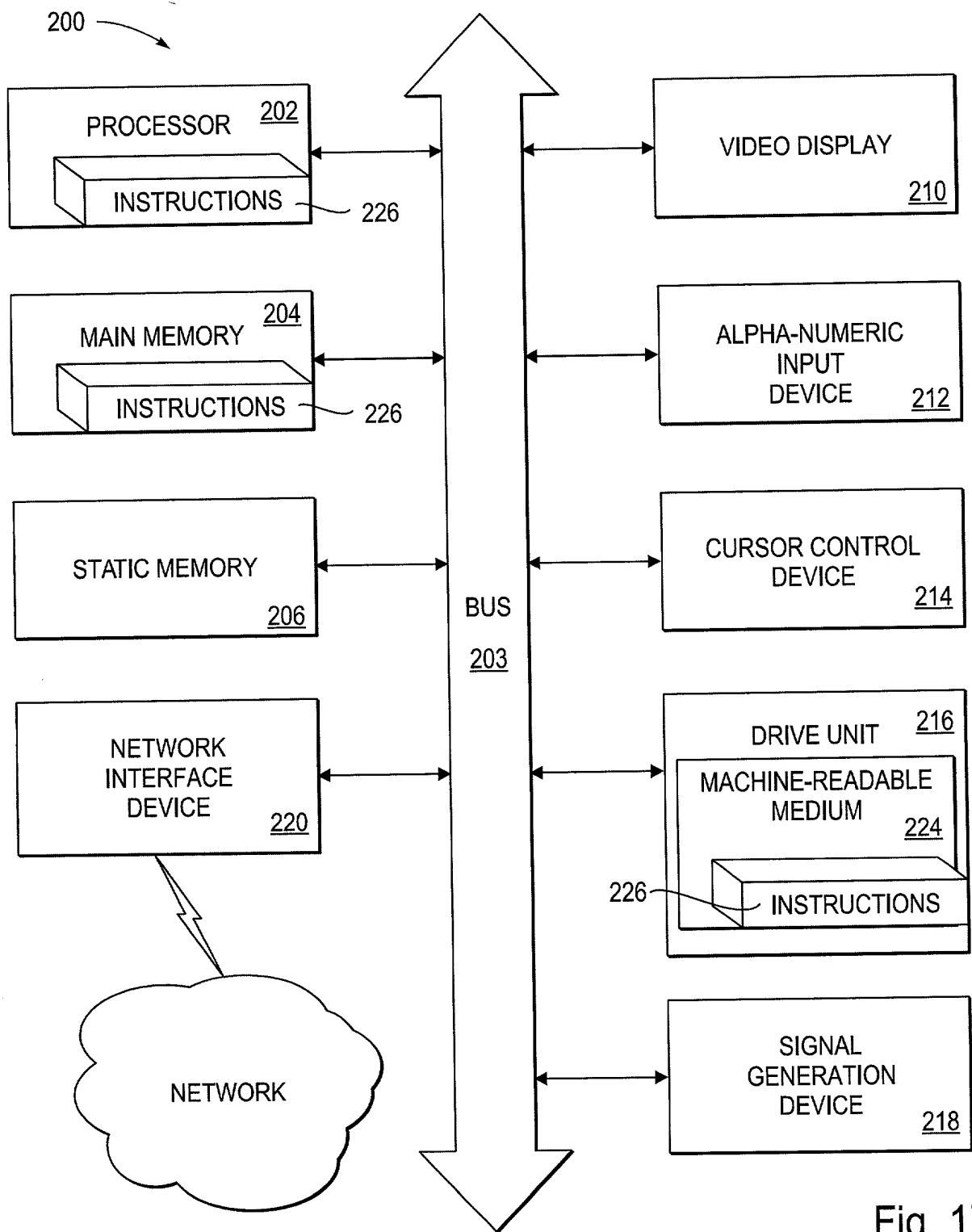


Fig. 17